

# Eclipse shallow synchronized MPI project at trestles

The gprof and gcov tools can be used with synchronized projects as well as local projects. These steps describe using the shallow project in synchronized form at trestles.sdsc.edu.

The latest Eclipse Juno has a couple gotchas to watch for:

- switch to the System Monitoring perspective and start the monitor for the machine you'll use before attempting to submit a job with Run Configurations, failure to start the monitor may cause the job submit to error out
- for gcov and gprof, the executable file has to be selected each time you open a .gcno or gmon.\* file (this seems to be unique to the synchronized project usage)

The contents of \$HOME/.bash\_profile should match up with the settings you choose from Project->Properties->C/C++Build->Environment\_Management

\$HOME/.bash\_profile :

```
module unload mvapich2
module unload pgc
module load gnu/4.6.1
module load openmpi/1.4.3
```

The decs.h file in the shallow project was modified to increase the itmax number of timesteps to 9000 so that the program runs slightly longer for performance analysis (it's still nearly instantaneous on a quick cluster node ).

To get the synchronize project code looking nice and clear of warnings, do the following for the trestles setup:

Project->Properties->C/C++ General->Code Analysis

- uncheck Symbol is not resolved
- uncheck Type cannot be resolved

C/C++ General->Paths & Symbols

- add //trestles.sdsc.edu/home/diag/opt/openmpi/1.4.3/gnu/include
- add //trestles.sdsc.edu/usr/include

Project->Index->Rebuild or Freshen All Files

Add the GMON\_OUT\_PREFIX to the run configuration and set it to gmon\_shallow or something similar.

Import a custom PBS script into the workspace on trestles or add it with an editor window:

```

#!/bin/bash
#PBS -A TG-STAl10022S
#PBS -q shallow
#PBS -N shallow
#PBS -l nodes=1:ppn=4
#PBS -l walltime=00:05:00
#PBS -V
MPI_ARGS="-np 4"
if [ "-np" == "${MPI_ARGS}" ] ; then
    MPI_ARGS=
fi

module unload mvapich2
module unload pgi
module load gnu/4.6.1
module load openmpi

MYSCREXE=${HOME}/shallow/shallow
cd $HOME/shallow

COMMAND=mpirun
#COMMAND=mpirun_rsh
if [ -n "${COMMAND}" ] ; then
    COMMAND="${COMMAND} ${MPI_ARGS} -hostfile ${PBS_NODEFILE}  ${MYSCREXE}  "
else
    COMMAND="${MYSCREXE}  "
fi
${COMMAND}

```

Then run the program.

The gmon\_shallow.\* files can be combined to a summary file with: gprof -s shallow gmon\_shallow\* . You can also rename a gmon\_shallow.NNNNN file to gmon.out to compare the gprof data for one rank with the gmon.sum summary (try to avoid the first PID which is rank zero because it's not as interesting as the others) .

*gcov is not supported at -O3 opt. level*

Interesting areas that seem to recur in the timing profile are:

- near calc.c: 49
- near tstep.f90: 73